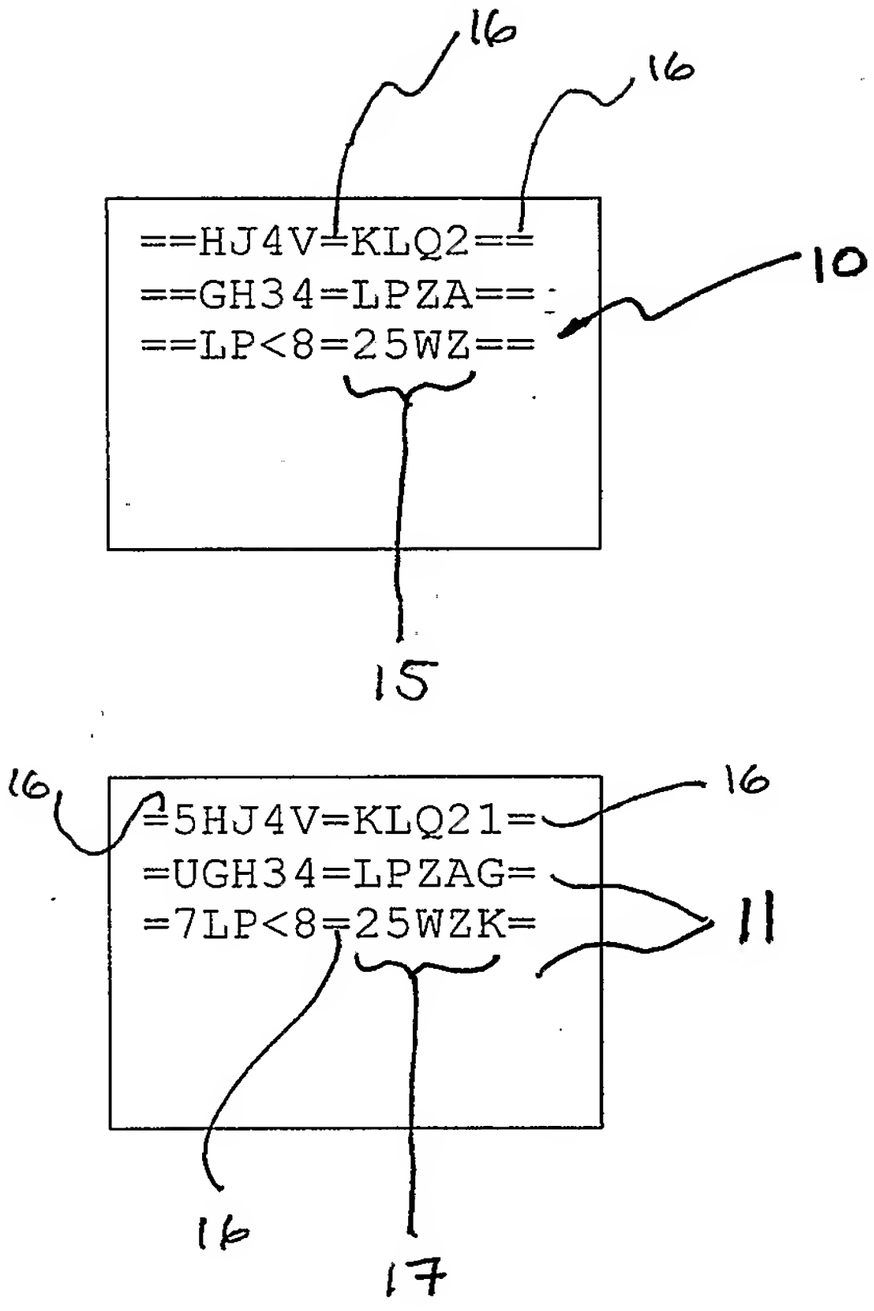

16
16
==HJ4V=KLQ2==
==GH34=LPZA==
==LP<8=25WZ==
10
15
16
16
=5HJ4V=KLQ21=
=UGH34=LPZAG=
=7LP<8=25WZK=
11
16
17

Fig. 1

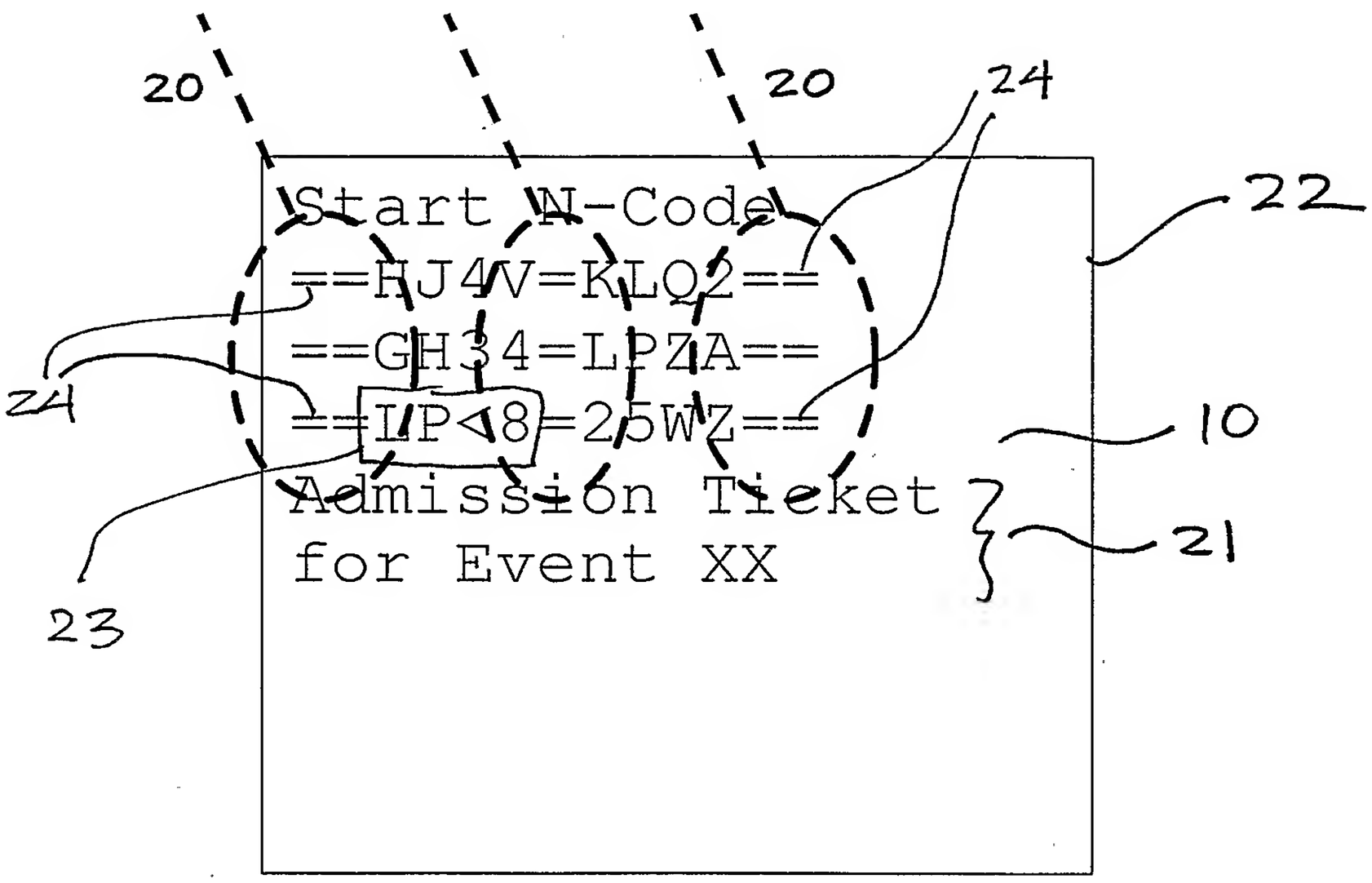

20
20
24
22
Start N-Code
==HJ4V=KLQ2==
==GH34=LPZA==
==LP48=25WZ==
Admission Ticket
for Event XX
24
10
21
23

Fig. 2

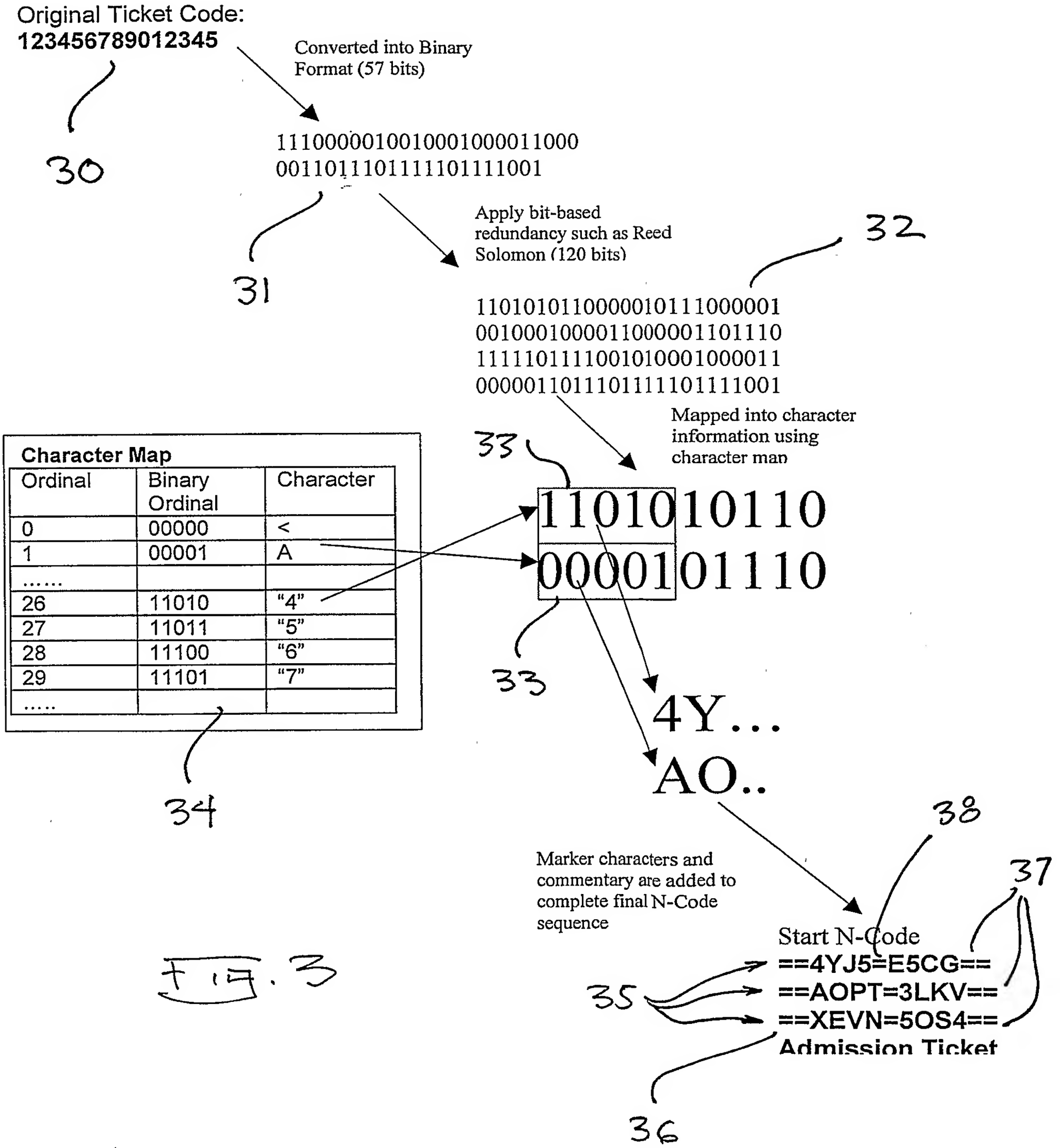
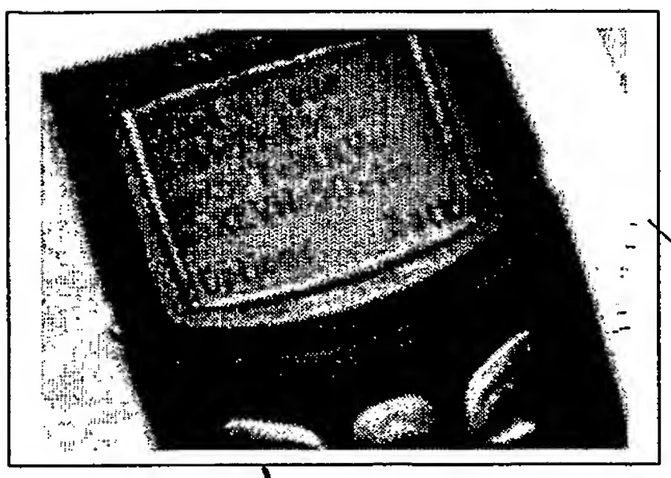

Original Ticket Code:
123456789012345
30
Converted into Binary Format (57 bits)
111000001001000100001100 0011011101111101111001
31
Apply bit-based redundancy such as Reed Solomon (120 bits)
32
110101011000001011100000 1 0010001000011000001101110 1111101111001010001000011 0000011011101111101111001
Mapped into character information using character map
33
Character Map
Ordinal | Binary Ordinal | Character
0 | 00000 | <
1 | 00001 | A
..... | |
26 | 11010 | "4"
27 | 11011 | "5"
28 | 11100 | "6"
29 | 11101 | "7"
..... | |
34
1101010110 0000101110
33
4Y...
AO..
38
37
Marker characters and commentary are added to complete final N-Code sequence
Start N-Code
==4YJ5=E5CG==
==AOPT=3LKV==
==XEVN=5OS4==
Admission Ticket
35
36

Fig. 3

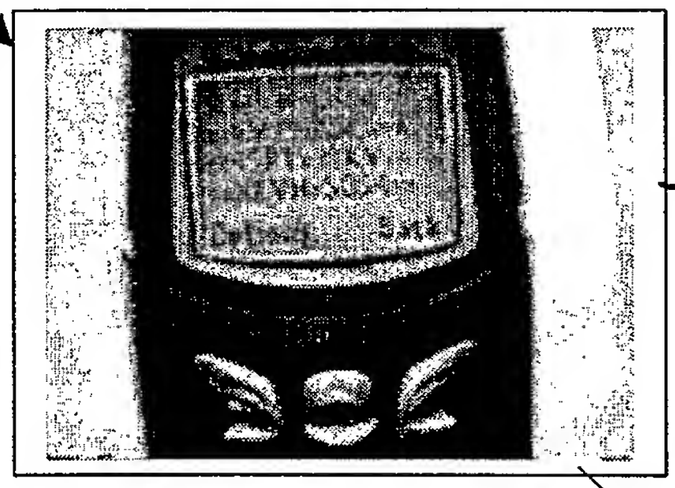
Fig. 4



42

De-skewing using middle screen data

41



43

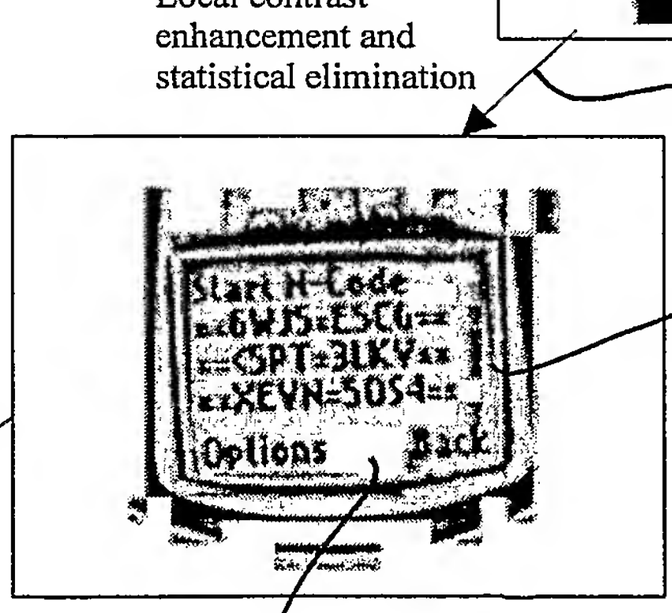
Global contrast enhancement

45

44

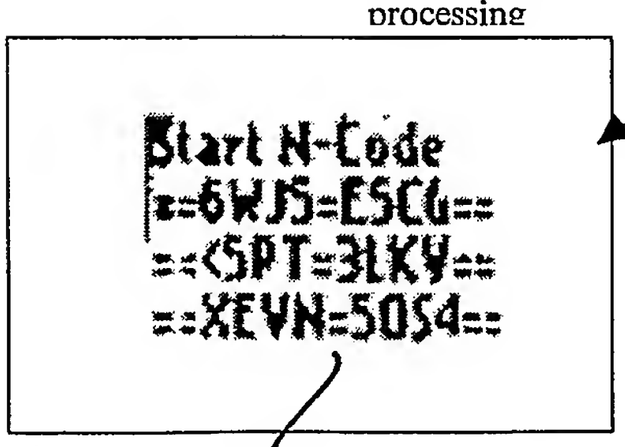
Local contrast enhancement and statistical elimination

46



47

Convert to black and white image for OCR processing



48

49

50

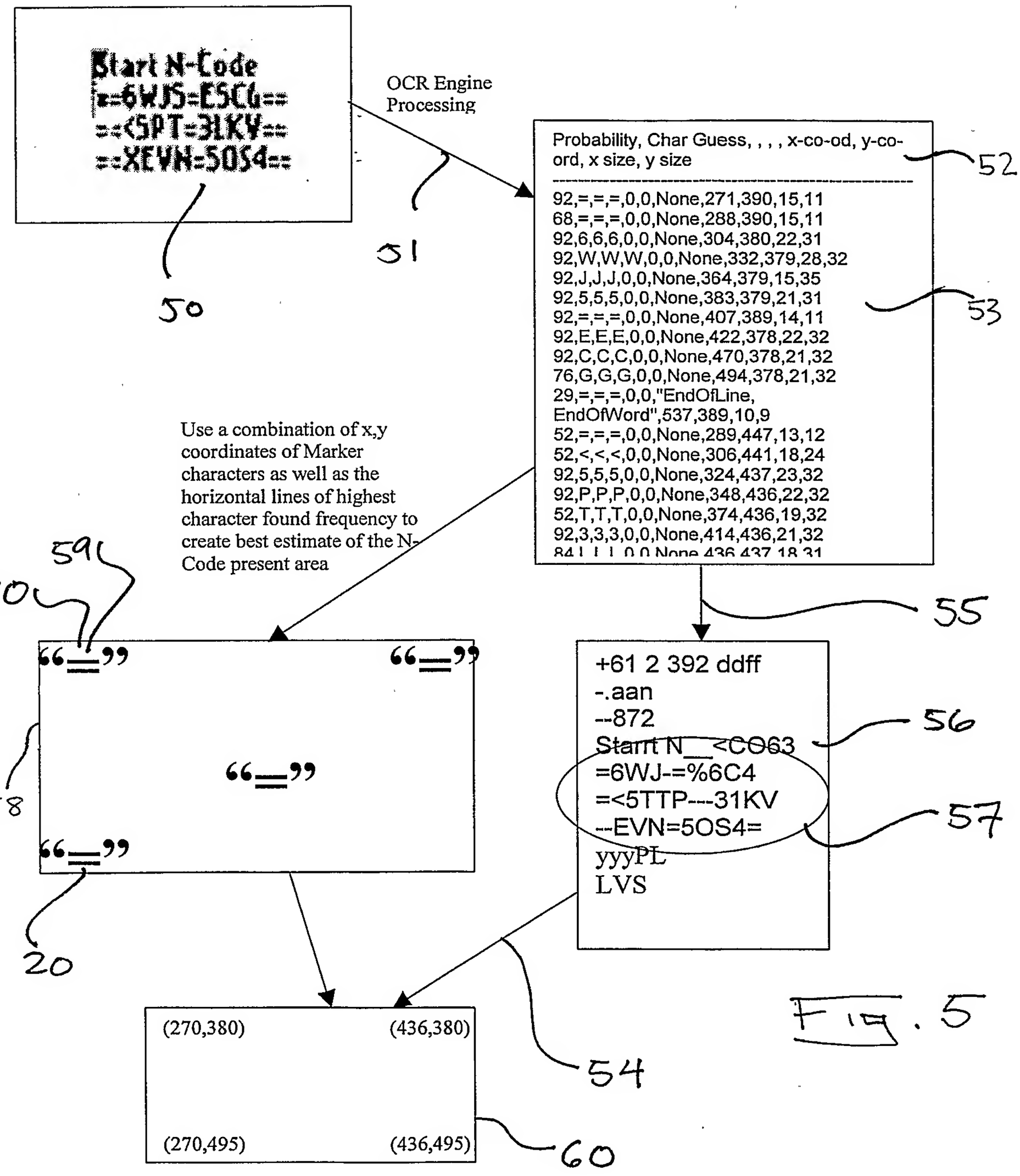

Start N-Code
=6WJ5=E5C6==
==<5PT=3LKV==
==XEVN=5OS4==
OCR Engine Processing
Probability, Char Guess, , , , x-co-od, y-co-ord, x size, y size
92,=,=,=,0,0,None,271,390,15,11
68,=,=,=,0,0,None,288,390,15,11
92,6,6,6,0,0,None,304,380,22,31
92,W,W,W,0,0,None,332,379,28,32
92,J,J,J,0,0,None,364,379,15,35
92,5,5,5,0,0,None,383,379,21,31
92,=,=,=,0,0,None,407,389,14,11
92,E,E,E,0,0,None,422,378,22,32
92,C,C,C,0,0,None,470,378,21,32
76,G,G,G,0,0,None,494,378,21,32
29,=,=,=,0,0,"EndOfLine, EndOfWord",537,389,10,9
52,=,=,=,0,0,None,289,447,13,12
52,<,<,<,0,0,None,306,441,18,24
92,5,5,5,0,0,None,324,437,23,32
92,P,P,P,0,0,None,348,436,22,32
52,T,T,T,0,0,None,374,436,19,32
92,3,3,3,0,0,None,414,436,21,32
50
51
52
53
55
Use a combination of x,y coordinates of Marker characters as well as the horizontal lines of highest character found frequency to create best estimate of the N-Code present area
59
20
58
20
+61 2 392 ddff
-.aan
--872
Start N__<CO63
=6WJ-=%6C4
=<5TTP---31KV
--EVN=5OS4=
yyyPL
LVS
56
57
(270,380)
(436,380)
(270,495)
(436,495)
54
60

Fig. 5

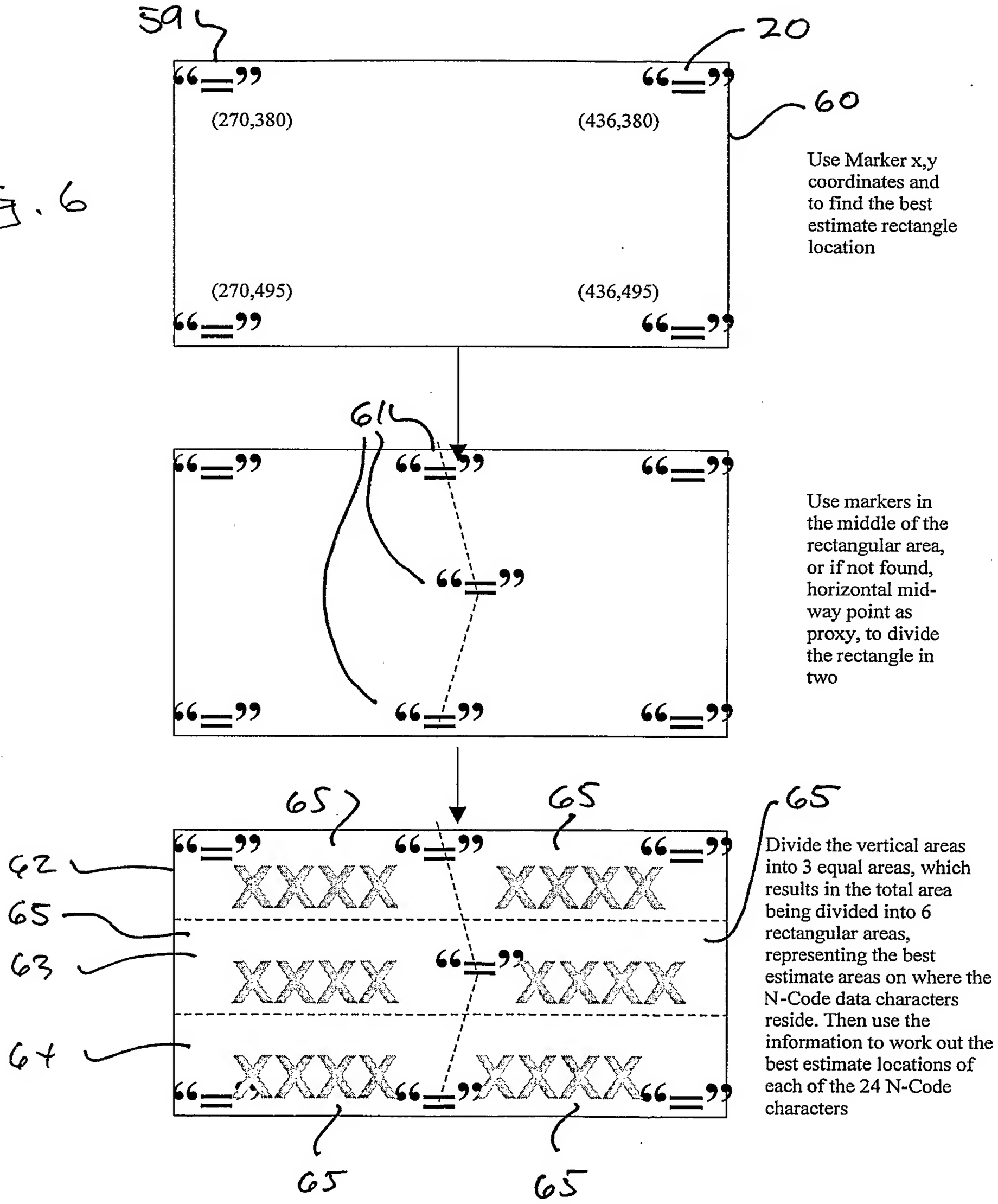
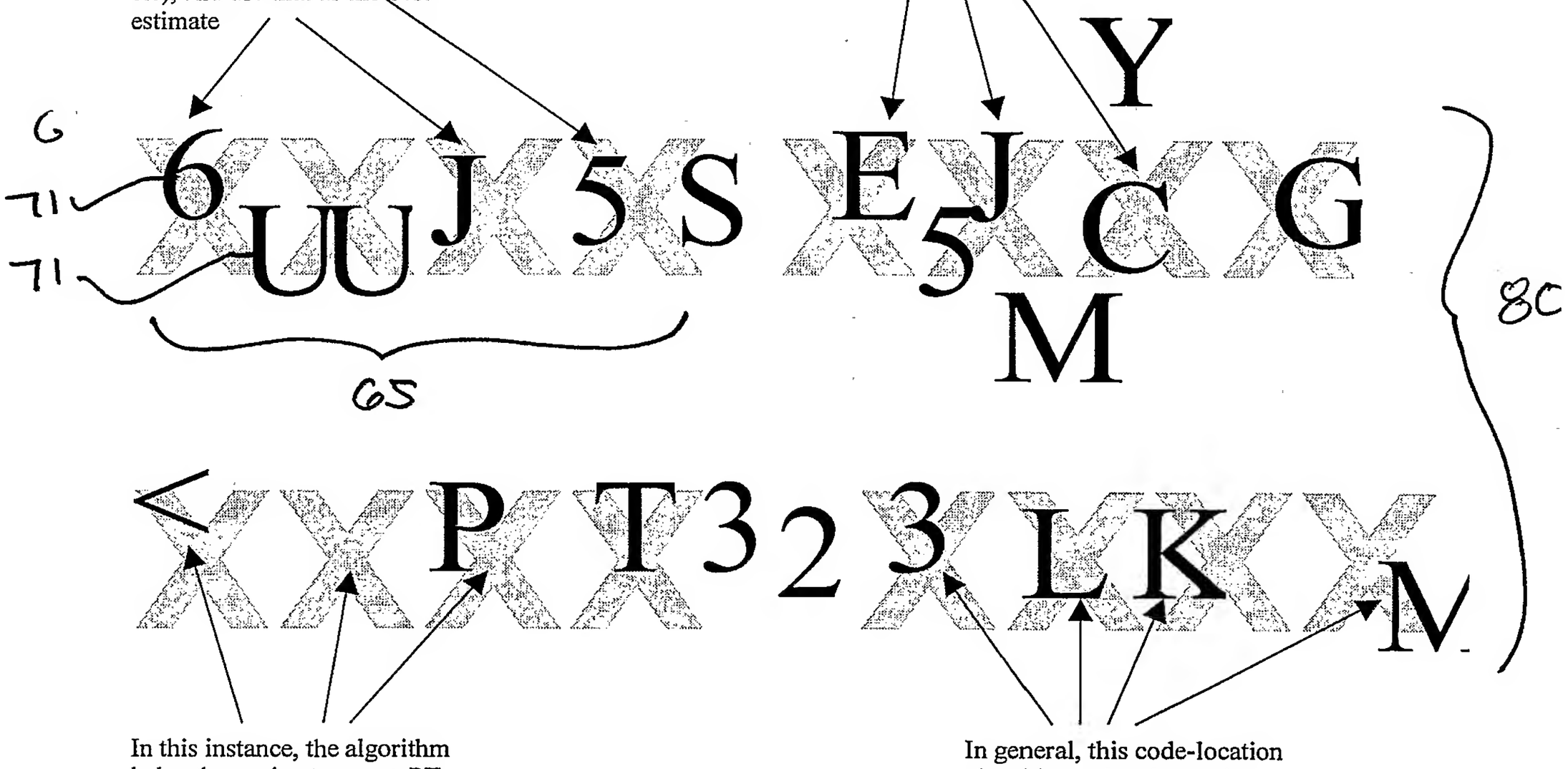

59
20
60
(270,380)
(436,380)
(270,495)
(436,495)
61
62
63
64
65
65
65
65
65
Use Marker x,y coordinates and to find the best estimate rectangle location
Use markers in the middle of the rectangular area, or if not found, horizontal mid-way point as proxy, to divide the rectangle in two
Divide the vertical areas into 3 equal areas, which results in the total area being divided into 6 rectangular areas, representing the best estimate areas on where the N-Code data characters reside. Then use the information to work out the best estimate locations of each of the 24 N-Code characters

Fig. 6

The algorithm determines the closest character returned by the OCR output (denoted by data characters starting with 6, U, etc) to each of the expected character positions (X, X, X, etc), and use that as the best estimate

In this instance, EJCG will be selected in favour of other surrounding "garbage" characters



In this instance, the algorithm helps the engine to use <_PT, and pass in the 2nd character as unknown, rather than using thinking <PT3 is the sequence. As <5PT is the actual sequence, this algorithm helps the engine know that it cannot determine what was in the second character, rather than getting 3 wrong with a <PT3 guess – as they will be misplaced

In general, this code-location algorithm helps the engine decide whether a character is there, or missing, and how likely is it to be correct, based on its distance away from the expected location. Also it helps the engine determine the best guess of the 4 data characters, rather than having to examine strings that are longer than 4 characters and be confused which 4 to pass on to the next step

Fig. 7

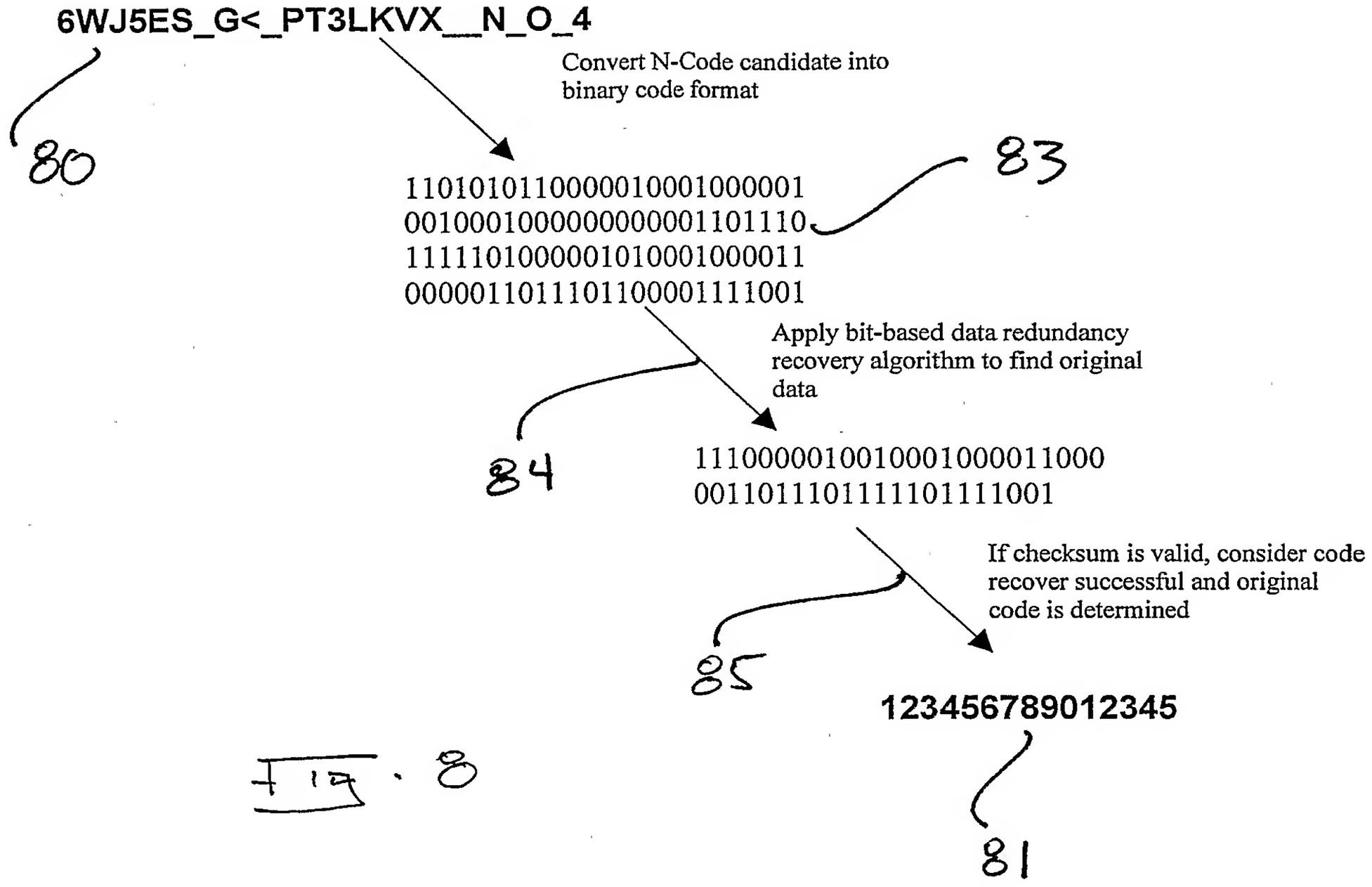

6WJ5ES_G<_PT3LKVX__N_O_4
80
Convert N-Code candidate into binary code format
83
11010101100000100010000010010001000000000001101110
11111010000010100010000110000011011101100001111001
Apply bit-based data redundancy recovery algorithm to find original data
84
11100000100100010000110000011011101111101111001
If checksum is valid, consider code recover successful and original code is determined
85
123456789012345
81

Fig. 8

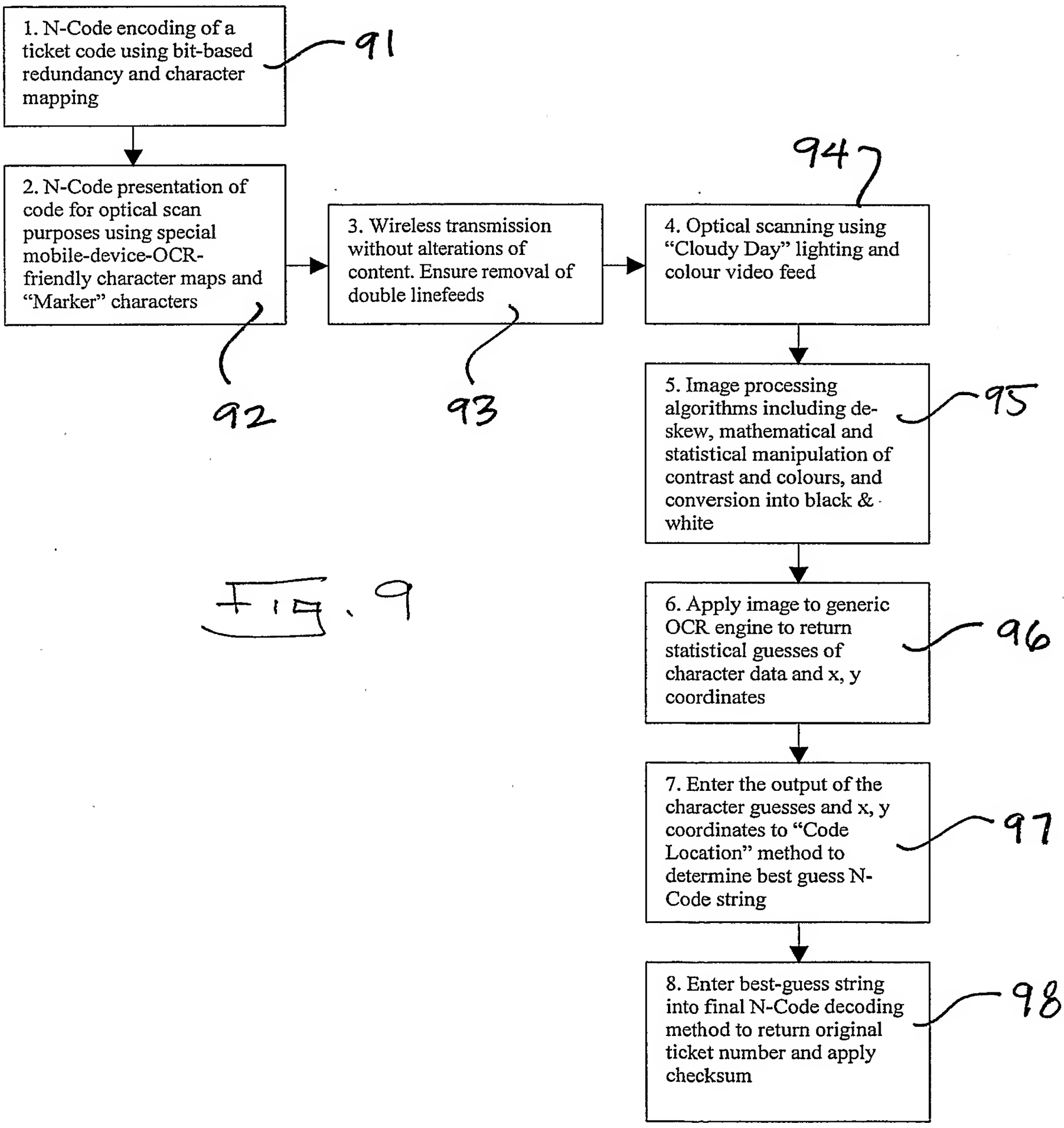

1. N-Code encoding of a ticket code using bit-based redundancy and character mapping
91
2. N-Code presentation of code for optical scan purposes using special mobile-device-OCR-friendly character maps and "Marker" characters
92
3. Wireless transmission without alterations of content. Ensure removal of double linefeeds
93
4. Optical scanning using "Cloudy Day" lighting and colour video feed
94
5. Image processing algorithms including de-skew, mathematical and statistical manipulation of contrast and colours, and conversion into black & white
95
6. Apply image to generic OCR engine to return statistical guesses of character data and x, y coordinates
96
7. Enter the output of the character guesses and x, y coordinates to "Code Location" method to determine best guess N-Code string
97
8. Enter best-guess string into final N-Code decoding method to return original ticket number and apply checksum
98

Fig. 9

Required 20-digit Data (102)	Temporary Mapped 12-Digit Data (101)
12345678901234567000	111222333000
12345678901234567001	111222333001
12345678901234567002	111222333002
12345678901234567003	111222333003
98765432109876543000	111222333004
98765432109876543001	111222333005
98765432109876543002	111222333006
98765432109876543003	111222333007
...	...

Mapping Table with Resell and Re-issue Requirement			
Required 20-digit Data (103)	Temporary Mapped 12-Digit Data	Active (Y/N?) (105)	Timestamp (106)
12345678901234567000	111222333000	Y	
12345678901234567001	111222333001	Y	
12345678901234567002	111222333002	Y	
12345678901234567003	111222333003	Y	
98765432109876543000	111222333004	Y	
98765432109876543001	111222333005	Y	
98765432109876543002	111222333006	N (104)	2004-10-12 12:34 (107)
98765432109876543002	111222333156	N	2004-10-14 10:34
98765432109876543002	111222333773	Y	2004-10-14 10:34
98765432109876543003	111222333007	Y	
...	...		

108

Fig 10

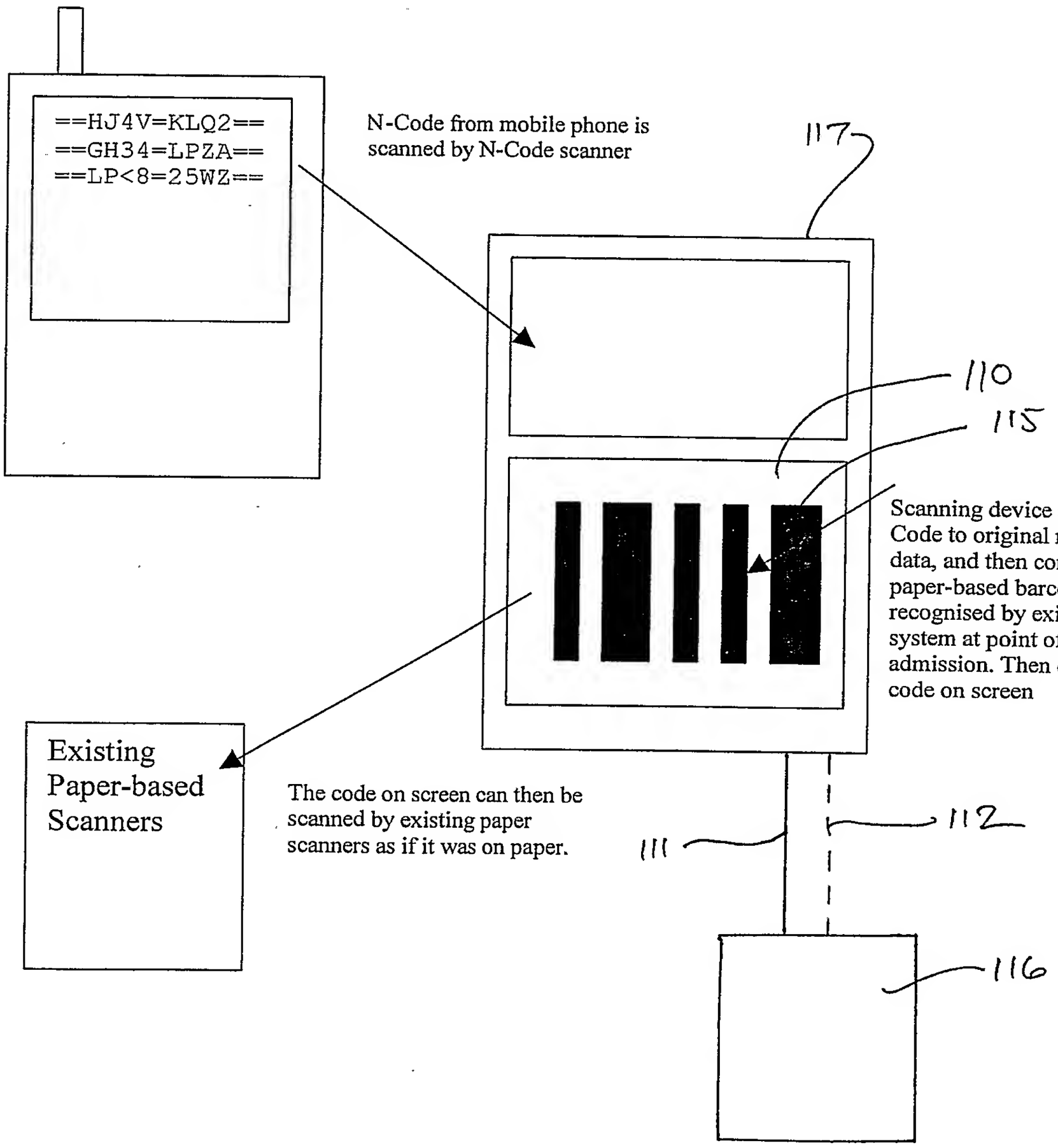

==HJ4V=KLQ2==
==GH34=LPZA==
==LP<8=25WZ==
N-Code from mobile phone is scanned by N-Code scanner
117
110
115
Scanning device converts N-Code to original numeric ticket data, and then converts that into paper-based barcode that is recognised by existing scanning system at point of sale or admission. Then displays this code on screen
Existing Paper-based Scanners
The code on screen can then be scanned by existing paper scanners as if it was on paper.
111
112
116

Fig. 11

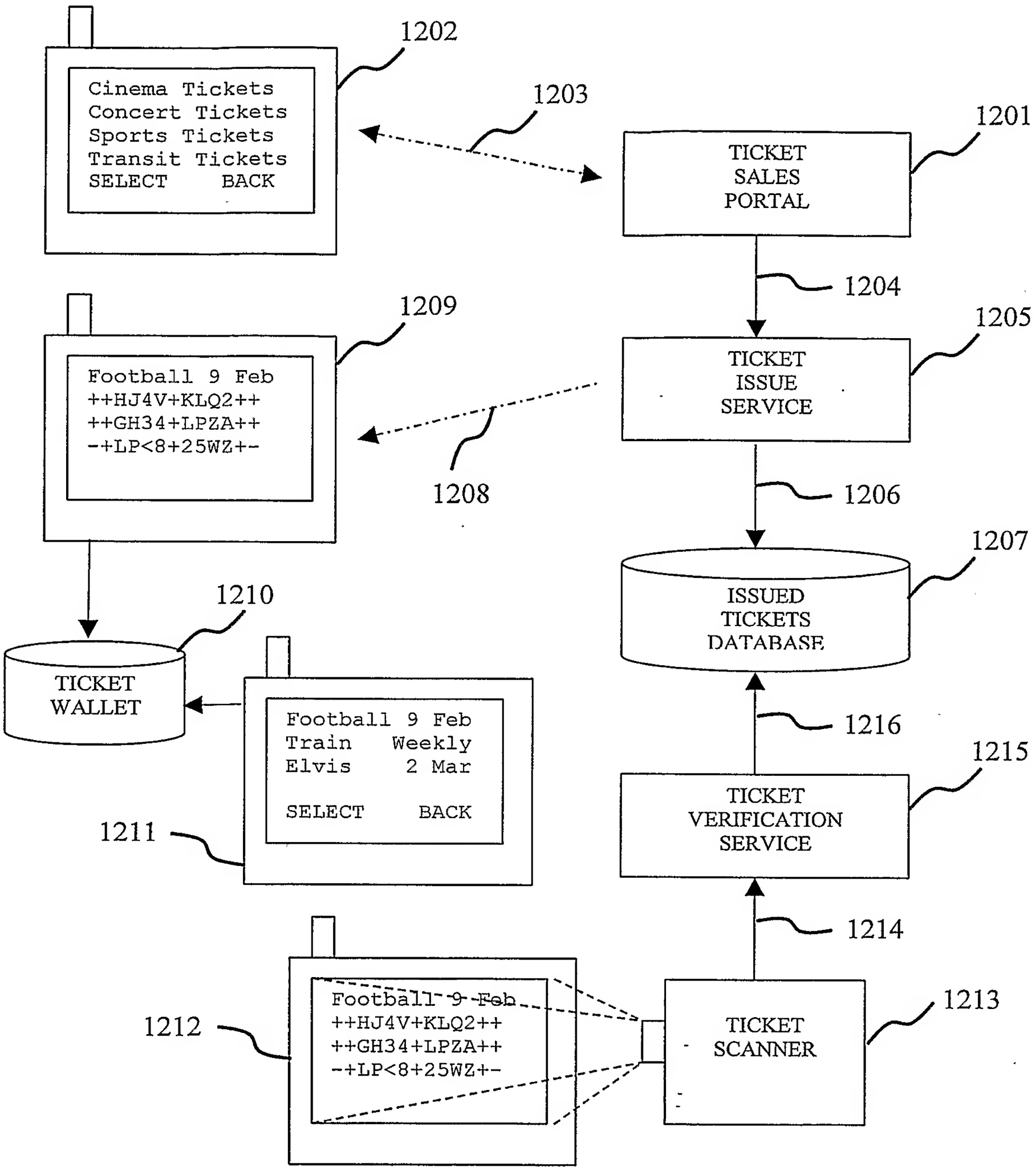

1202
Cinema Tickets
Concert Tickets
Sports Tickets
Transit Tickets
SELECT BACK
1203
1201
TICKET SALES PORTAL
1204
1205
TICKET ISSUE SERVICE
1209
Football 9 Feb
++HJ4V+KLQ2++
++GH34+LPZA++
-+LP<8+25WZ+-
1208
1206
1207
ISSUED TICKETS DATABASE
1210
TICKET WALLET
Football 9 Feb
Train Weekly
Elvis 2 Mar
SELECT BACK
1211
1216
1215
TICKET VERIFICATION SERVICE
1214
1212
Football 9 Feb
++HJ4V+KLQ2++
++GH34+LPZA++
-+LP<8+25WZ+-
1213
TICKET SCANNER

Fig 12